DOCUMENT RESUME

ED 453 285 TM 032 824

AUTHOR VanLehn, Kurt

TITLE Olae: A Bayesian Performance Assessment for Complex Problem Solving.

PUB DATE 2001-04-13

NOTE 18p.; Paper presented at the Annual Meeting of the National Council on Measurement in Education (Seattle, WA, April 11-13, 2001).

PUB TYPE Information Analyses (070) -- Reports - Descriptive (141) -- Speeches/Meeting Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.

DESCRIPTORS *Bayesian Statistics; *Computer Assisted Testing; Intelligent Tutoring Systems; Knowledge Level; Performance Based Assessment; *Physics; Problem Solving; *Rating Scales; Reliability; Student Evaluation; Validity

ABSTRACT

Olae is a computer system for assessing student knowledge of physics, and Newtonian mechanics in particular, using performance data collected while students solve complex problems. Although originally designed as a stand-alone system, it has also been used as part of the Andes intelligent tutoring system. Like many other performance assessment systems, Olae compares a student's problem-solving behavior step-by-step with the behavior of an ideal model solving the same problem. The main feature of Olae is use of Bayesian networks to assign credit to pieces of knowledge when the student makes a correct step; and blame to knowledge pieces when the student makes an incorrect step. This paper introduces the basic principles of Olae and illustrates how it solves the classic assignment of credit and blame in a way that is not only mathematically sound by also intuitively satisfying. The paper then reviews a series of evaluations that measured the reliability of Olae and its validity and sensitivity to its parameters. This paper synthesizes results across these studies to support the conclusion that, although Olae is a viable solution to the problem of complex performance assessment, all model-based assessments have two fundamental inadequacies that cause them to lose important data if students stop following expected solution paths or refuse to enter their intermediate results into the computer. These inadequacies harm their accuracy no matter what method of data analysis is used. In fact, human assessors, who were used as the gold standard when evaluating Olae, are equally hampered by these inadequacies. A promising solution appears to be to use tutoring systems to do assessment. (Contains 3 figures and 19 references.) (Author/SLD)

Reproductions supplied by EDRS are the best that can be made from the original document.

ED 453 285

Olae: A Bayesian performance assessment for complex problem solving

Kurt VanLehn
CIRCLE: Center for Interdisciplinary Research on Constructive Learning Environments
Learning Research and Development Center
University of Pittsburgh, Pittsburgh, PA 15260
http://www.pitt.edu/~vanlehn
Copyright © Kurt VanLehn 2001

Presented at the National Conference on Measurement in Education, April 13, 2001, Seattle, WA

PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL HAS BEEN GRANTED BY

K. VanLehn

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

1

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☑ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

TM032824

BEST COPY AVAILABLE

Olae: A Bayesian performance assessment for complex problem solving

Kurt VanLehn
CIRCLE: Center for Interdisciplinary Research on Constructive Learning Environments
Learning Research and Development Center
University of Pittsburgh, Pittsburgh, PA 15260
http://www.pitt.edu/~vanlehn
Copyright © Kurt VanLehn 2001

Presented at the National Conference on Measurement in Education, April 13, 2001, Seattle, WA

Abstract

Olae is a computer system for assessing student knowledge of physics, and Newtonian mechanics in particular, using performance data collected while students solve complex problems. Although originally designed as a stand-alone assessment system, it has also been used as part of the Andes intelligent tutoring system. Like many other performance assessment systems, Olae compares the student's problem solving behavior step-by-step to the behavior of an ideal model that solving the same problem. The main feature of Olae is its use of Bayesian networks to assign credit to pieces of knowledge when the student makes a correct step and blame to knowledge pieces when the student makes an incorrect step. This paper begins by introducing the basic principles of Olae and illustrating how it solves this classic assignment of credit and blame problem in a way that is not only mathematically sound but intuitively satisfying. It then reviews a series of evaluations that measured its reliability, its validity and its sensitivity to its parameters. Although earlier articles presented this work in full detail, this paper synthesizes results across studies to support the following conclusions: Although Olae is indeed a viable solution to the problem of complex performance assessment, all model-based assessments have two fundamental inadequacies that cause them to lose important data. These inadequacies harms their accuracy no matter what method of data analysis is used. Indeed, human assessors, which were used as the gold standard when evaluating Olae, are equally hampered by these inadequacies. A promising solution appears to be to use tutoring systems to do assessment.

What is Olae?

From the student's point of view, Olae is a physics problem solving environment that is as unconstraining and unresponsive as a piece of paper. Olae presents physics problems such as, "A block slides down a frictionless plane that is inclined at 37 degrees. What is its acceleration?" Solving such a problem typically involves drawing a body, drawing forces, drawing a coordinate system, writing equations and solving them algebraically. The student can do all such work on Olae's user interface, which provides tools for each of the actions just listed. The user interface tries to put no other constraints on the student. Students can show all their work or only some of it. The order of actions is unconstrained as well. Olae gives no feedback on students' actions, nor on the final answer. It is an assessment instrument, and not a tutoring system. The name "Olae" is an acronym for "On-line assessment of expertise."

Olae has also been used as the student modeling component of the Andes physics tutoring system (Gertner & VanLehn, 2000). Figure 1 shows the Andes screen. Students are presented with a physics problem in

the upper left window. They draw vectors, axes, etc. in that window as well. They define variables in the upper right window and they enter equations in the lower right window. Whenever they make a correct entry, it turns green; incorrect entries turn red. When they ask for help, hints appear in the lower left window. The Olae screen looks essentially the same, except that entries do not turn red or green, and there is no way to request help.

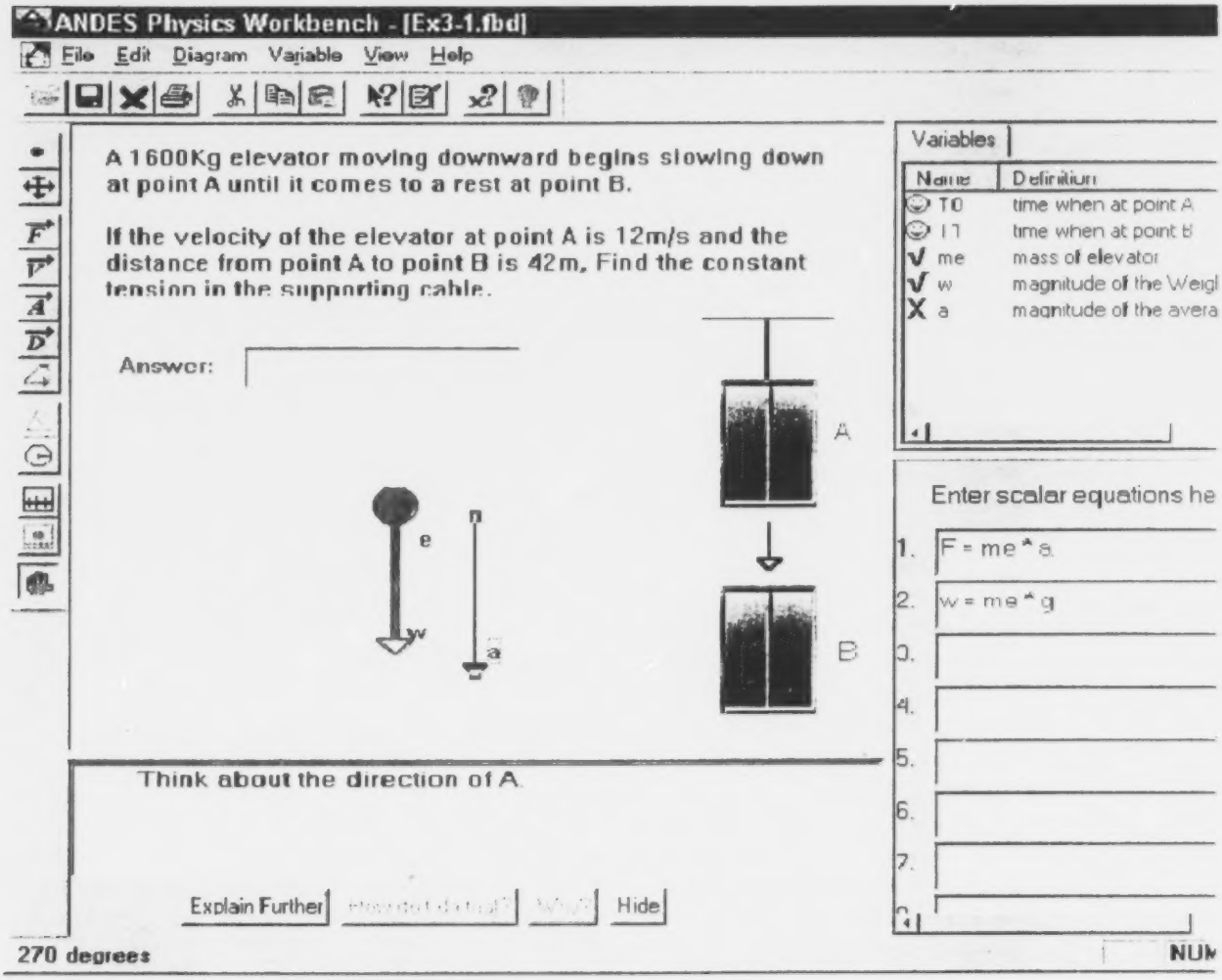
ANDES Physics Workbench - [Ex3-1.fbd]
File Edit Diagram Variable View Help
A 1600Kg elevator moving downward begins slowing down at point A until it comes to a rest at point B.
If the velocity of the elevator at point A is 12m/s and the distance from point A to point B is 42m, Find the constant tension in the supporting cable.
Answer:
A
B
e
n
w
a
Variables
Name Definition
T0 time when at point A
T1 time when at point B
me mass of elevator
w magnitude of the Weigl
a magnitude of the avera
Enter scalar equations he
1. F = me * a
2. w = me * g
3.
4.
5.
6.
7.
Think about the direction of A.
Explain Further
Hide
270 degrees
NUM

Figure 1 The Andes intelligent tutoring system

From an assessment point of view, Olae addresses the problem of performance assessment of complex problem solving. This is an important problem due to the recent stress on complex problem solving in standards and in teaching of both traditional content domains (e.g., NCTM) as well as vocational and professions domains. The hope is that if tests use complex, intrinsically important problems, and schools adapt their instruction to the tests, then their instruction will improve (Linn, Baker & Dunbar, 1991).

Olae's problems are complex in that they take many minutes to solve, and they involve instinsically important physics concepts such as "force" and "acceleration" that are notoriously prone to difficult-to-remedy misunderstandings. Such problems are often used for homework, but physics exams tend to use multiple-choice questions instead. Because the problems take so long to solve, an exam could use only 2 or 3 problems, which means that the grader would have to consider the derivation of the answers

BEST COPY AVAILABLE

as well as the answers themselves. However, understanding students' derivations from their scratch work is difficult because the steps of the derivation may be carried out in many different orders, and students may choose to do some steps in their head instead of writing them down. These features make it difficult and unreliable for human graders to score derivations of physics problems. This is typical of complex problem solving. Olae provides a solution to grading complex problem solving that is reliable, valid and completely objective.

From a technical point of view, Olae uses a unique combination of probabilistic and rule-based technology. Olae has a knowledge base consisting of Prolog-like rules. The rules can solve all the problems given by Olae to students. Each rule represents a small unit of physics knowledge, such as "Any object that is near a planet has a gravitational force acting on it due to the planet," or "An object that is moving with constant velocity has zero acceleration." Solving a problem typically requires 50 to 200 rule applications. Although most rules represent correct knowledge, some represent common misconceptions and other incorrect beliefs. The latter are called "buggy rules."

In order to prepare for analyzing student work on a particular problem, Olae first produces all possible solutions to the problem using its knowledge base. The number of possible solutions is roughly proportional to the number of ways to order the rule applications, which means that there are 50! to 200! possible sequences of rule applications. Olae stores the rule applications in a partial order (a directed acyclic graph), which compactly represents all possible sequences. This solution graph, as it is called, also keeps track of the propositions that were accessed as premises of the rule applications and the propositions that were produced a conclusions. The solution graph is stored in a file.

When the student arrives, Olae poses the problem to the student, loads the solution graph file, and monitors the student's actions while solving the problem. The basic idea is that when a student's action matches a rule application node in the solution graph, Olae infer that the student must know the corresponding rule. Moreover, the student must also know the rules that produced the facts used as premises of this rule application, because the student probably would apply the rule only if its premises were available.

However, such inferences are necessarily probabilistic, for several reasons. First it might be that the student's action resulted from a guess or some other form of reasoning that is not represented by rule applications. Second, it might be that there are several rule applications that correspond to the student's actions, thus making it unclear which one should receive credit for the student's action. For instance, a buggy rule and a correct rule may produce exactly the same conclusions on some problems even though they produce different conclusions on other problems. In order to handle such probabilistic reasoning in a sound manner, the solution graph is converted into a Bayesian network (Russell & Norvig, 1995). Although Bayesian networks are fairly widely used now, Olae was one of the first assessment systems to use them.

The basic idea is that when there are multiple interpretations for a student's action, the Bayesian network assigns credit in direct proportion to the prior probabilities of the interpretation. For instance, if there are two interpretations for an action, say a correct rule C and a buggy rule B, and the correct rule was much more likely to be known by the student than the buggy rule, then the network assigns more credit to C than B according to Bayes rule: $P(C|A)/P(B|A) = P(A|C)P(C) / P(A|B)P(B)$. Here $P(C)$ is the probability of knowing rule C before seeing the student's action, A, $P(C|A)$ is the probability of knowing rule C after seeing the student's action A, and $P(A|C)$ is the probability of the student doing action A if they know rule C. Similarly, $P(B)$ is the prior probability of knowing rule B, $P(B|A)$ is the posterior probability of knowing rule B, and $P(A|B)$ is the probability of doing action A if rule B is known. Suppose we assume that all rules are equally likely to apply when they are known (this is not exactly what Olae assumes, but it is close enough). This means that $P(A|C)=P(A|B)$, so we obtain $P(C|A)/P(B|A) = P(C)/P(B)$. In other

words, more credit (posterior probability) is assigned to rules with more prior probability.

This may at first glance seem unfair (the rich get richer!), but it actually makes intuitive sense. If you observed a correct action that could also be generated by a rare bug, you'd be more likely to assume that student knew the correct rule rather than the rare buggy one. It is just using frequency distributions of interpretations in the whole population to bias our interpretation of an individual action. Nonetheless, there are deep philosophical issues here that have been debated for decades.

In actual practice, it is not simple to assign credit to rules when actions are observed. First, one must consider not only the rule applications that correspond directly to the action, but all the rule applications that produced the propositions that are used as *premises* of the target rule applications. If it is likely that one of them has not occurred, then the requisite premise is probably missing, so the rule application cannot occur. Second, one must worry about the *conclusions* produced by the target rule applications. If one of them is supposed to produce a certain conclusion, and that conclusion leads to other rule applications none of which have been observed, then it is likely that the conclusion was never drawn and thus the rule application never occurred. Even when just one action is observed, its interpretation can potentially involve every rule application in the network. This means that observation of a single action can potentially change the posterior probabilities of every rule application in the network.

In general, updating a Bayesian network when new actions are observed is an NP-hard problem, which means that the time required to update the posterior probabilities in the network can rise exponentially with the number of nodes in the network. Nonetheless, Olae is able to analyze even its largest Bayesian networks in a few minutes or less. Martin and VanLehn (1995) give a complete description of Olae.

An example of how Olae assigns credit and blame

This section illustrates the operation of Olae on a very simple problem. Figure 2 shows the solution as a Bayesian network. The problem has two given propositions, A and B. The premise of Rule 1 matches Proposition A and the resulting rule application produces Proposition C, which matches the premise of Rule 3, and its application produces Proposition E. Similarly, applying Rule 2 to Proposition B produces Proposition D, and applying Rule 4 to D produce a second derivation of Proposition E. Let us assume that the two given propositions are observable in that they are printed in the problem statement. Let us also assume that the student has just one user interface action available, which is to enter an answer, which may or may not match Proposition E. Thus, Propositions C and D are unobservable, because there are no user interfaces action that the student can make which represent them. In Figure 2, the observable propositions are shaded and the unobservable ones are not.

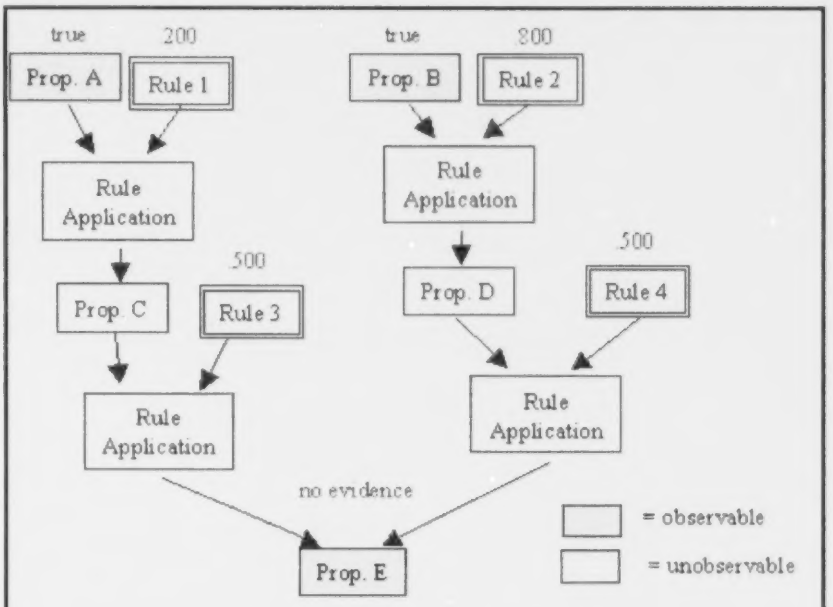

Figure 2: Olae's Bayesian network for a simple problem's solution

When Olae loads in the solution graph file, it initializes a Bayesian network with the prior probabilities of the rules. In this case, lets assume the prior probabilities of rules 1 through 4 are .2, .8, .5 and .5 respectively (shown above the rule nodes in Figure 2). Nodes that have parents are assigned conditional probabilities that reflect a simple cognitive theory of reasoning:

- In order for a rule to apply, it must be mastered and all its premises much match a proposition that is in working memory. Thus, P(rule applies | some premise is not in working memory or rule is not mastered) = 0.
- Even if a rule can apply, then there is still some chance that the conclusion will not be drawn. This is called the "slip" parameter, and is typically set to .001. Thus, P(rule applies | all premises are in working memory and rule is mastered) = 0.999
- If a rule applies, then its conclusion is in working memory. Thus, P(proposition in working memory | some parent rule application occurs) = 1.0.
- If a proposition cannot be derived via rule applications, there is still some chance that it will be obtained by guessing or other means. This is modeled with the "guess" parameter, which is typically around 0.2. Thus, P(proposition in working memory | no parent rule application occurs) = 0.2.

Next Olae clamps the given propositions to true, which represents the assumption that the student has those propositions in working memory as a result of reading the problem statement. All the other observable nodes (there is just one in this case, Proposition E) are clamped neither true nor false, since the student has not yet entered any user interface actions. Figure 2 shows the Bayesian network in this state.

Suppose that the student now enters a user interface action that corresponds to Proposition E. That is, the student gets the solution of this trivial problem correct. In order to assign credit, Olae clamps the nodes corresponding to the user interface actions (there is just one, namely proposition E) to true, then updates the network. This causes the posterior marginal probabilities on the rule nodes to rise as shown in Figure 3. To see if this makes intuitive sense, note that the students action can be explained in two ways: either the student applied rules 1 and 3 or the student applied rules 2 and 4. Which explanation should receive the most credit? The prior probabilities of the right-side rules are equal to are higher than the priors on the left,

so it is a priori the more likely derivation of the solution. When the student actually does solve the problem, the right-side explanation gets more credit, just as Bayes rules says that it should.

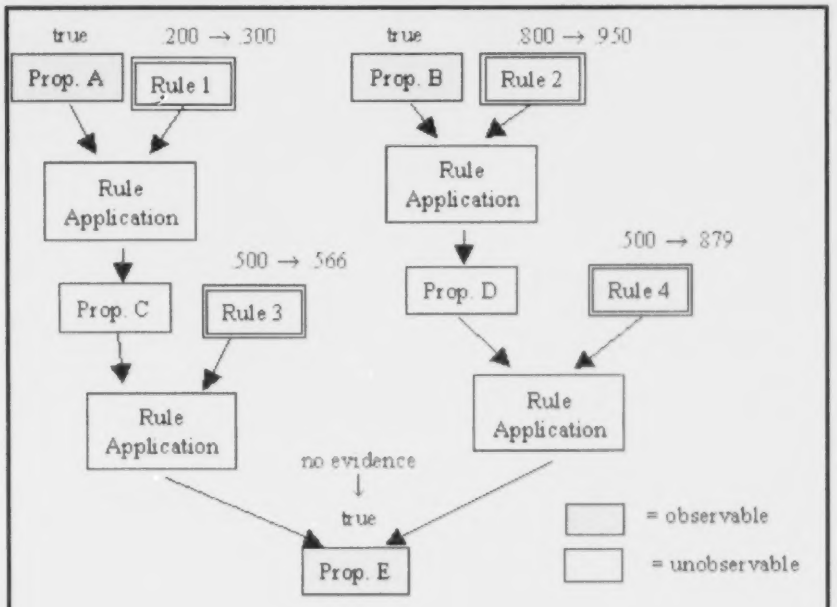

Figure 3: Credit for doing a step flows from the evidence towards the more probable explanation

Now consider what happens if the student enters an incorrect solution to the problem. In this case, the student clearly does not have Proposition E in working memory, so Olae clamps it false and updates the network. Figure 4 shows the result. Both derivations have been blamed. However, Rules 3 and 4 received more blame than rules 1 and 2. This makes sense; if a chain of rules is known to be broken somewhere, then it is probably the weakest link that broke.

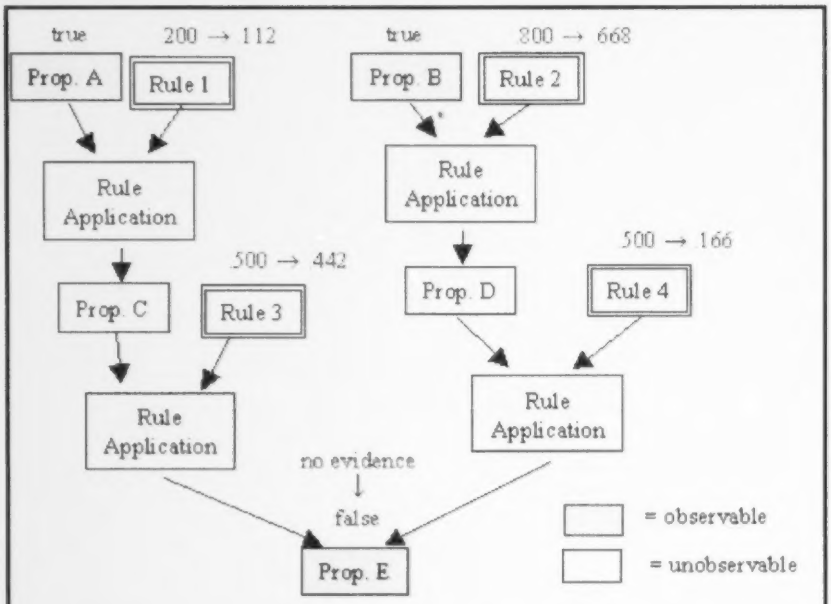

Figure 4: Blame for missing a step flows from the evidence towards the less probable steps in both explanations

This illustrates how Olae assigns credit and blame. It is a mathematically sound, principled solution to the assignment of credit and blame problem. This is a significant advance over the heuristic solutions used in prior work. See Jameson (1995) for a review that compares Olae to other numerical methods and to earlier, heuristic approaches.

Calibration

Calibration is the process of obtaining estimates for the values of the assessor's parameters. Olae has several kinds of parameters. The prior probabilities of the rules are the parameters whose values are most uncertain and most in need of empirical estimation. General theories of cognition constrain the other parameters, such as the probability that a student will "slip" and not apply a rule that is known even when all its premises are present. Using subjective estimates is perhaps more acceptable for the constrained parameters.

One calibration procedure for estimating prior probabilities of rules is based on the traditional EM (Estimation/Maximization) technique, which is a form of hill climbing. The first step is to generate a random assignment of probabilities. The next step is to run Olae on the subjects in the sample, and obtain their assessment. From these assessments, each rule's frequency in the sample can be calculated. The next step is to revise the prior probabilities to reflect the frequency distribution in the sample, then run Olae on each subject again. This gives us a new and presumably better estimate of the rules' frequencies in the population. The next step is to again adjust the priors and run Olae. This continues until there are no significant changes in the priors. From a hill-climbing perspective, the process has reached a local maximum. The next phase is to repeat the whole process again, starting with a different random assignment of prior probabilities. This eventuates in a new local maximum. This is repeated many times, keeping track of the local maxima reached each time, where each maximum is an assignment of prior probabilities. After many runs, the procedure accepts the most popular local maximum as a global maximum. In this fashion, the EM procedure obtain the "best" setting of prior probabilities. In principle, the EM procedure can be used for setting all the parameters in Olae.

The parameter values obtained via EM are good estimates of the population parameters only if the sample of students is large. When Olae was evaluated in 1997, a large sample of student behavior was not available, so the evaluation that is described in the next section was conducted with subjective estimates for parameter values. A later analysis (also described below) measured the sensitivity of Olae to its parameter values.

Another method of calibration was developed later. A pencil and paper test was constructed where each rule in the knowledge base was tested by its own item. Although this is probably not as good as the EM procedure, it allowed estimation of prior rule probabilities from a smaller sample of students. The test and its scoring procedure are described by VanLehn, Niu, Siler and Gertner (1998).

The reliability Olae

The reliability of a test is often evaluated by partitioning its items to form two "subtests" (e.g., the even items and the odd items), then seeing if students' scores on one subtest correlate with their scores on the other subtest. This procedure is difficult to apply to Olae because it assesses competence at a fine grain size. That is, it reports about 200 probabilities, one for each rule in its knowledge base. This creates a small technical problem, which is that one must combine those 200 numbers into a single aggregate "score." Although the score is not meaningful, it can be used in the correlation that determines reliability.

The real problem is that an Olae test has only a few problems on it, because each of those problem is so complex that students take a long a long time to solve them. If one creates subtests of the odd and even problems, then they will undoubtedly involve different sets of rules. This will unfairly depress the correlation. In order to evaluate reliability fairly, one would have to construct a test consisting of pairs of isomorphic problems (e.g., same physics, different cover story). This could be a rather long test, and there is a risk that students will detect the isomorphisms. Any assessment of complex problem solving has this same problem in trying to use the traditional method for evaluating reliability.

One solution is to use a measure, *predictive accuracy*, that is widely used for systems that induce models from data (Russell & Norvig, 1995). The evaluation method begins by dividing the data into two parts, called the training data and the test data. The to-be-evaluated system induces a model from the training data. The model makes predictions about the testing data. Comparing these predictions to the actual testing data establishes the predictive accuracy of the data analyzer. Typically, this process is repeated with many different partitions of the data into training and test sets.

To evaluate the predictive accuracy of Olae, each student's test was analyzed as follows. The first step was to generate rule probabilities using data from all but one of the student's problems. The second step is to use the rule probabilities to predict the students performance on the remaining problem, (Olae can both calculate student actions given rule probabilities and calculate rule probabilities given student actions.) In particular, Olae predicted exactly which vectors and equations the student should write given the assessed rule probabilities. The third step was to compare the predicted actions to the observed actions. These three steps were repeated once for each problem, using it at the test problem and the other problems as the training problems.

The predictive accuracy method is similar to the traditional method for measuring reliability, except that it uses data from N-1 problems to predict performance on the Nth problem. This makes it more likely that every rule in the Nth problem is used at least once in the other problems. This means that a lack of predictive accuracy can be blamed on Olae, and not on the lack of overlap between problems.

Unfortunately, predictive accuracy is a new way to measure reliability, so no one knows whether the measured predictive accuracy (0.90) is good enough. More experience with the measure is necessary. Measuring the predictive accuracy of existing, well accepted assessments would be a good start. VanLehn and Martin (1998) describe the evaluation of Olae's reliability more thoroughly.

The validity of Olae

Although reliability can be assessed without even knowing the meaning of a test score, validity measures the "veracity" or "appropriateness" of a test relative to its intended use. There are many ways to measure validity (Messick, 1989), and several were applied to Olae.

A common one is criteria-related validity, which asks, "do test scores correlate well with other measures of the target competence?" The criteria-related validity of Olae was determined as follows. In earlier work, a rule-based model of physics cognition, named Cascade, was fit to 9 subjects (VanLehn & Jones, 1993; VanLehn, Jones, & Chi, 1992). This fitting was based on verbal protocols as well as the worksheets of the subjects. For each of the subjects, the vectors and equations that the subject wrote on the worksheet were entered into Olae so that it could assess the student. Thus, the human assessors had access to more data than Olae, because they had the verbal protocols as well as the worksheets. Thus, their assessment could be considered a "gold standard."

The match between the human assessment and Olae's assessment seemed good. In all cases where the human assessors determined that the student knew a rule, Olae assigned a probability of greater than 0.85 to the rule. In all cases where the human assessors determined that the student did not know a rule, Olae assigned a probability of less than 0.15. That is, Olae and the human assessors were in 100% agreement.

In order to determine if this degree of match could occur by chance, a "random" assessment was developed. It decided if a rule is mastered by flipping a weighted coin. The coin's weight was the proportion of students mastering that rule. For instance, if only 1 of the 9 students had mastered rule A, then the coin's weight for rule A would be 1/9. Suppose that Olae's assessment matched the human assessment on rule A for all 9 subjects. How likely is it that such a perfect match would be generated randomly, by flipping the weighted coin? For a student who had not mastered the rule, according to the human assessment, there is a 8/9 chance of the coin indicating non-mastery, and for the student who had mastered the rule, there was a 1/9 chance of the coin indicating mastery. Since there are 8 students who had not mastered the rule and one who had, the chance of the coin generating a perfect match to the human assessment is $[(8/9)^8](1/9)=.043$. Thus, it is unlikely ($p<.05$) that Olae's perfect match to the human assessment is due to chance. On the other hand, suppose that all students had completely mastered rule B, and Olae also estimated that all students had mastered rule B. In this case, the weighted coin always picks mastery, so the chance of Olae matching the human assessors for all 9 subjects on rule B is $(9/9)^9=1.0$. That is, if all the student have mastered a particular rule, then the fact that Olae correctly predicts the rule's mastery is quite unimpressive.

Calculated in this fashion, the probability that Olae's assessment would agree with the human assessment on all 25 rules turned out to be 0.000006. Clearly, Olae's perfect agreement with the human assessors was no fluke.

These findings suggest that Olae's criteria-related validity is high, but there are other kinds of validity to consider as well. *Content validity* is usually assessed by having experts rate whether the items on a test are representative of tasks in the target domain. In the case of Olae, the problems were drawn from textbooks, so presumably they are typical in both form and content. However, it is possible to select problems that only use a few of the many rules available in a domain. To measure content validity more objectively, one

could extend Olae's knowledge base to handle all problems in a textbook, then see what proportion of those rules are required to solve the particular problems that appear on a test. This can actually be done, because Olae's knowledge base has now been extended as part of the Andes tutoring system, and can now solve almost all mechanics problems assigned in physics classes at the US Naval Academy (Gertner & VanLehn, 2000). Of course, this would only measure content validity relative to complex problem solving, and that is only part of what physics courses teach. Olae currently does not, for instance, monitor the students' ability to carry out experiments or analyze experimental data.

One of the unusual aspects of Olae is its complex "scoring method" (i.e., its use of Bayesian networks, etc.). This makes one wonder if it could be a source of invalidity. To evaluate Olae's scoring method, a variant on criteria-related validity was applied. Instead of using human assessment as a gold standard, "simulated students" that could solve Olae's physics problems were employed. Because we constructed the students, we knew exactly which rules they had mastered. Using this method, Olae once again performed well relative to a null model. The details of all these validity evaluations, as well as reflections on other kinds of validity, are in VanLehn and Martin (1998).

The sensitivity of Olae

It is important to know how sensitive Olae is to its parameters. For instance, if varying the prior probabilities on rules does not affect the accuracy of Olae's assessments, then one can worry less about getting a large sample of students for use in estimating those parameters values. Although VanLehn and Niu (2001) discuss the sensitivity analysis in detail, the main results will be summarized here.

In order to measure the assessment accuracy, simulated students were again used to generate "performance data." This allowed repeated measurements of accuracy with many kinds of students, such as high vs. low competence students, or students that guessed frequently vs. rarely. This facilitated understanding how varying a parameter's value affected Olae's accuracy for different kinds of students.

First, it turned out that for most student actions, there was no assignment of credit problem. When a correct action was observed, there was almost always just one explanation for it. The two-explanation situation shown in Figure 2 turned out to be rare. In fact, only 3% of the proposition nodes were like node E in Figure 2 in that they had more than one parent. This meant that varying the values of the prior probabilities made little difference in the assignment of credit (except as noted below). Digging deeper, this turned out to be due to two factors. At the time of the sensitivity analysis, Olae had been transformed into a module, called *the assessor*, of the Andes physics tutoring system. Andes had little use for buggy rules, as it had a different method for diagnosing errors which was simpler to implement (Gertner, 1998). Consequently, the knowledge base used in the sensitivity assessment had no buggy rules in it. When there were no buggy rules, the only way to have multiple explanations of a correct action was for that action to play a role in multiple correct solutions. Most of the problems used in the analysis had only one solution, although students could order the solution's steps in combinatorially many ways. Consequently, whenever a correct action was observed, there was no question about which rules had to be involved in the corresponding rule application, and so exactly those rules got all the credit. Thus, the two factors that caused the assessor to be insensitive to its priors were (a) the knowledge base had no buggy rules in it, and (b) the problems generally had just one correct solution.

The second finding involved the assessor's *guess* and *slip* parameters. These parameters modified the production of conclusions from rule applications. Ideally, a rule application occurs and produces its conclusion if and only if the rule is known and all the rule's premises are known. However, humans are not ideal reasoners, so it was assumed that a rule's conclusion might occasionally (with P = the slip parameter) *not* be produced even though the rule and the premises are known. It also assumed that the conclusion

generated by a rule might occasionally (with P = the guess parameter) be produced even if the rule application does not occur.

The assessors' only parameters were the prior probabilities of the rules, the guess parameters and the slip parameters, and it turned out there were important interactions among them. If the ratio of guess parameters to prior probabilities was high, then the assessor tended to think that conclusions were generated by guessing. Thus, it took much positive evidence (student actions) before the posterior probability of a rule began to rise above its prior probability. Similarly, if the ratio of the slip parameters to the prior probabilities was high, then the assessor tended to think that missing actions were due to slips rather than lack of knowledge. Thus, it took much negative evidence (missing student actions) before the posterior probabilities of the rules began to fall below the prior probabilities.

The assessor turned out to be sensitive to its prior probabilities, but not in the way anticipated. If rule A had a higher probability than rule B, then positive evidence would make rule A's posterior probability rise faster than rule B's because the assessor thought that the student guessed less often with rule A than with rule B. Similarly, negative evidence would make rule B's posterior fall faster than rule A's. This occurs even if the two rules never completed to explain an action.

The necessity of keeping students moving along a solution path

As mentioned earlier, Olae's assessment technique is now being used in the context of the Andes tutoring system. Andes gives immediate feedback after each student action by coloring incorrect actions red and correct actions green. Students can also ask for help. For an overview of Andes, see Gertner and VanLehn (2000). The papers cited there provide details.

Intuitively, trying to assess student knowledge while the student is being tutored should cause lower accuracy, because the student's competence may change while the assessment is being conducted. Thus, the sensitivity analysis included varying whether or not feedback was given in order to determine its effects on accuracy. During this initial study of the relationship between feedback and assessment accuracy, no attempt was made to simulate student learning during the assessment. The simulated students were set up so that they would have the same set of known rules regardless of how much feedback they got. However, the feedback did have the effect of correcting their mistakes and putting them back on a solution path. Thus, the study simulated only the benefits of feedback for assessment without simulating its drawbacks. It also did not simulate student requests for help, nor Andes' response.

The result, which is described further in VanLehn and Niu (2001), is that feedback dramatically increased the assessment accuracy. It turned out that when the simulated students did not get feedback, they stopped moving along a solution path. That is, they either got stuck or they made an error that put them on an incorrect, garden path. Either way, they no longer produced actions that the assessor could interpret. Because the assessor had only correct rules, the solution graph contained only correct rule applications, and those would match only correct actions. Thus, once a student had stopped following a solution path, the subsequent student actions (if any) could not be analyzed by the assessor. Even if the subsequent rule applications were produced by correct rule applications, they were applied to incorrect premises and thus produced incorrect conclusions. Thus, many correct rule applications would not receive the credit they deserved.

In particular, when the simulated students had low competence and feedback was turned off, they stopped following the solution path so early that there were hardly any correct actions for the assessor to recognize. Even when such students solved 23 physics problems, their assessed competence was well below their actual competence. These students' competence was constructed to approximate the competence of human

students who were just beginning a physics course, as determined by with the short-item calibration test mentioned earlier (VanLehn, Niu, Siler and Gertner 1998).

Any method of assessment for complex problem solving that can only recognize correct actions will have the same problem as this assessor. It must either use feedback to keep students moving along a solution path or it must have some means of recognizing correct rule applications that occur after the student has left a solution path.

In particular, the human assessors whose performance was compared to Olae's as a measure of criteria-related validity could not give feedback. If the student went off the solution path, then the assessors might be able to follow the student's reasoning but they may also have found some solutions just as impenetrable as Olae did. In short, it is likely that both the human assessors and Olae failed to credit students with competence on some rules just because the student applied those rules in a context the where the assessors could not recognize the rules' applications.

There are two known computational techniques for recognizing correct rule applications that are applied to incorrect propositions and thus produce incorrect conclusions. The first method, pioneered by Brown and Burton (1978), is simply to augment the knowledge base with many buggy rules. This enables the assessor to recognize most incorrect actions as well as all correct ones. Unfortunately, collecting a large set of buggy rules requires examining data from many students.

The second method, pioneered by Burton and Brown (1982) and refined by Mitrovic and Ohlsson (1999), represent each student action as a pair of states: the state just before the action and the state just after the action. Every rule is tested against every pair. If the rule's premises match the state prior to the action and its conclusions match the state after the action, then the rule is given credit for applying. If the rule's premises match the state before the action but its conclusions do not match the state after the action, then it is blamed for not applying. This method solves the problem because a rule will get credit for applying even if the premises in the prior state are incorrect. Unfortunately, it requires that the states before and after each rule application be observable by the assessor, which is often not easily achieved.

It seems then that assessment of complex problem solving might best be done by tutoring systems that give immediate feedback or in some other way constrain students to stay on solution paths. However, before endorsing this method, another study should be done. The simulated students should learn from the feedback, and the assessor should be modified to expect that learning will occur. For instance, the assessor might use a moving window: only evidence from the most recent N problems is used to assess competence (Gitomer, Steinberg & Mislevy, 1995). Although this makes sense in a tutoring situation, in a testing situation, it is equivalent to limiting the test to just N complex problems. An alternative modification is to equip the assessor with a simple model of learning and forgetting, such as the knowledge tracing technique of the Anderson et al. (1995) tutors, or the somewhat fancier version of Shute (1995) that takes levels of hints into account. With a modified assessor and modified simulated students, one could study the tradeoff between the benefits of immediate feedback (it increases the number of recognizable actions) and its costs (it changes student's competence during the assessment).

Requiring students to show their work

Another finding from the analysis of VanLehn and Niu (2001) is that assessment accuracy is dramatically improved if students are required to show their work. That is, for specified types of actions, students are required to enter the action if they can. Thus, the absence of such actions means that the student cannot do them, which allows the assessor to propagate blame (negative evidence) through the Bayesian network.

Olae and Andes did not require students to show their work. For instance, although tools were provided for drawing vectors and coordinate axes, students were not required to use them. If they could envision the vector drawing, that was just fine. This policy was adopted by Olae because it was trying to be as unconstraining as a piece of paper. The policy was adopted by Andes because it was thought to increase the likelihood that students would use Andes voluntarily.

However, the sensitivity analysis found that assessment accuracy was harmed by letting students envision intermediate results (or write them on scratch paper, out of sight of the computer). If students are not required to show their work, then there is only one action that is required, and that is the final action of entering an answer. Thus, blame is propagated only when an incorrect answer is entered or the student quits, leaving the problem unanswered. At that point, so many rules are involved in producing the correct answer that spreading blame among them hardly affects their posterior probabilities at all. In fact, if students are not required to show their work, there is so little negative evidence that the posterior probabilities of all rules, regardless of whether the student actually knows them or not, gradually rise. This harms the assessor's accuracy.

In principle, the prior probabilities of rules should have a strong effect on the assignment of blame. The rules with low prior probabilities should absorb more blame when negative evidence is entered. Since very little negative evidence is available to the assessor, it was insensitive to its prior probabilities not only when interpreting correct actions, as shown earlier, but when interpreting *all* actions.

On the one hand, this is good news, because it means that we don't need accurate estimates of the prior probabilities of rules. On the other hand, it is bad news, because it suggests that there is no point in having prior probabilities at all. The assessment is essentially non-Bayesian, except for the competition between rules, guesses and slips which could presumably be handled in a non-Bayesian fashion equally well. To put it more generally, when assessing complex problem solving, there appears to be no advantage for Bayesian methods over non-Bayesian methods if (a) the assessor does not require students to show their work, (b) the assessor has no buggy rules, and (c) the task domain rarely produces multiple correct derivations of a correct action.

Open questions and future work

Although one open question has been mentioned already (the costs and benefits of immediate feedback), there are many more that deserve investigation. This section sketches a few that were raised by the Olae investigations.

The sensitivity analyses found that Olae was insensitive to the prior probabilities of its rules, and thus its use of Bayesian networks was probably not worth the effort. However, a number of improvements to Olae were suggested, such as using feedback and requiring students to show work. It is not clear how sensitive an improved Olae would be to its prior probabilities. We need to find this out in order to determine whether the Bayesian network approach is worth its inherent computational complexity.

Olae is based on a well known technique from intelligent tutoring systems, called *model tracing* (Anderson et al., 1995). Assessors based on model tracing interpret the student's actions by manipulating a model so that it will also generate those actions. When the student's actions are fine grained, such as writing a single vector or equation, then the rules in the model are necessarily rather fine grained as well. Thus, the assessment is also fine grained. In the case of Olae, the assessment produces a vector of probabilities, one number per rule, for hundreds of rules. Tutoring systems can use such fine grained assessments, but it is not clear what else they are good for.

Assessments should be thought of as decision aids: they help a person or machine decide which course of action to take. As a general rule of thumb, the longer the course of action, the larger the granularity of the assessment required to make a decision about it. If you want to decide whether to take Physics 2, then you need to know your grade for all of Physics 1. If you want to decide whether it is time to go on to chapter 10, then you need to know whether you have mastered chapter 9. If you want to know which problem to try next, then you need to know which rules you have not yet mastered and which rules each problem requires. That is, as the duration of the proposed course of action shrinks from a year down to minutes, the grain size of the requisite assessment shrinks from a whole course down to individual rules. This rule of thumb suggests that the fine grained assessments produced by model tracing assessors will be directly useful only when making small decisions.

However, it is easy to define a large grained assessment in terms of a fine grained ones. The original version of Olae contained a facility for users to draw Bayesian networks that defined larger grained objectives, such as "Newton's first law mastery" or "chapter 9 mastery", in terms of rule level mastery. The open issue is whether such aggregated assessments are more reliable or valid than other methods of forming the same large grained assessments. Clearly, this depends just as much on the aggregation method as it does on the reliability and validity of the rule level assessor. This whole area needs exploration.

Conclusions

Despite all the problems that have been observed for Olae and its successor, the Andes assessor module, the whole enterprise of basing assessments on complex problem solving is still sound. Complex problem solving is closer to authentic tasks than the short answer and multiple-choice items of conventional tests, so using complex problems for assessment is probably more beneficial to schools than conventional assessment methods (Linn, Baker & Dunbar, 1991). Moreover, the problem of monitoring a student's activity during complex problem solving and forming an assessment based on hundreds or thousands of student actions has been solved. Olae performs a mathematically sound assignment of credit and blame, and it is computationally tractable. Moreover, it does just as well has human graders who had more information than it did.

However, even though the credit/blame assignment problem has been solved, other problems remain. No amount of mathematically sophisticated analysis will overcome incomplete data, and important data are lost if students stop following expected solution paths or refuse to enter their intermediate results into the computer. One possible solution is to use tutoring systems for assessment, as they can put students back on solution paths and require them to show their work. Moreover, research is needed on how to appropriately aggregate rule-level assessments into larger-grained, more widely useful assessments. When these hurdles are cleared, it should be possible to not only replace the legion of graduate students and instructors who grade homework papers, but it should also be possible to use complex problem solving for higher stakes assessments.

References

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive Tutors: Lessons Learned. *The Journal of the Learning Sciences*, 4(2), 167-207.

Brown, J. S., & Burton, R. (1978). Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science*, 2, 155-192.

Burton, R. R., & Brown, J. S. (1982). An investigation of computer coaching for informal learning activities. In D. Sleeman. and J. S. Brown (Ed.), *Intelligent Tutoring Systems*. New York: Academic Press.

Gertner, A.S. (1998) Providing feedback to equation entries in an intelligent tutoring system for Physics. (pp. 254-263) In Goettle, B.P., Halff, H.M., Redfield, C.L. & Shute, V.J. (Eds). *Intelligent Tutoring Systems: Proceedings of the 4th International Conference on Intelligent Tutoring Systems*. Berlin: Springer. http://www.pitt.edu/~vanlehn/distrib/Papers/Gertner-ITS98.ps

Gertner, A. & VanLehn, K (2000). Andes: A Coached Problem Solving Environment for Physics. (pp. 131-142) In Gauthier, G., Frasson, C. & VanLehn, K. (Eds) *Intelligent Tutoring Systems: 5th International Conference, ITS 2000*, Berlin: Springer. http://www.pitt.edu/~vanlehn/distrib/gertnerabs.html

Gitomer, D. H., Steinberg, L. S., & Mislevy, R. J. (1995). Diagnostic assessment of trouble-shooting skill in an intelligent tutoring system. In P. Nichols & S. Chipman & R. Brennan (Eds.), *Cognitively diagnostic assessment*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Jameson, A. (1995). Numerical uncertainty management in user and student modeling: An overview of systems and issues. *User Modeling and User-Adapted Interaction*, 5 (3/4):193-251. http://www.wkap.nl/oasis.htm/103543

Linn, R. L., Baker, E. L. & Dunbar, S.B. (1991). Complex, performance-based assessment: Expectations and validation criteria. *Educational Researcher*, 20(8), 15-21.

Martin, J. & VanLehn, K. (1995) Student assessment using Bayesian nets. *International Journal of Human-Computer Studies*, Vol. 42., pp. 575-591. Academic Press. http://www.pitt.edu/~vanlehn/distrib/journal/HCS95-abs.html

Messick, S. (1989). Validity. In R. L. Linn (Ed.), Educational Measurement (3 ed., pp. 13-103). New York: Macmillan.

Mitrovic, A., & Ohlsson, S. (1999). Evaluation of a constraint-based tutor for a database language. *International Journal of Artificial Intelligence and Education*, 10, 238-256.

Russell, S., & Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Los Altos, CA: Morgan-Kaufman.

Shute, V.J. (1995). SMART: Student modeling approach for responsive tutoring. *User Modeling and User-Adapted Interaction*. 5(1): 1-44. http://www.wkap.nl/oasis.htm/86441

VanLehn, K., & Jones, R. M. (1993). Learning by explaining examples to oneself: A computational model. In S. Chipman & A. L. Meyrowitz (Eds.), Foundations of knowledge acquisition: Cognitive models of complex learning (pp. 25- 82). Boston: Kluwer. http://www.pitt.edu/~vanlehn/distrib/Chipman93-abstract.html

VanLehn, K., Jones, R. M., & Chi, M. T. H. (1992). A model of the self-explanation effect. Journal of the Learning Sciences, 2(1), 1-60. http://www.pitt.edu/~vanlehn/distrib/JLS92-abstract.html

VanLehn, K. & Martin, J. (1998) Evaluation on an assessment system based on Bayesian student modeling. *International Journal of Artificial Intelligence and Education*, 8(2), 179-221. http://www.pitt.edu/~vanlehn/distrib/journal/EvalAssess-abs.html

VanLehn, K. & Niu, Z. (2001). Bayesian student modeling, user interfaces and feedback: A sensitivity

analysis. *International Journal of Artificial Intelligence in Education*. 12, printed version to appear.
http://www.pitt.edu/~vanlehn/distrib/journal/SensitivityAnalAbs.htm

VanLehn, K., Niu, Z., Siler, S. & Gertner A. (1998) Student modeling from conventional test data : A Bayesian approach without priors. In: Proceedings of the 4th Intelligent Tutoring Systems ITS'98 Conference. Springer-Verlag Berlin Heidelberg, 1998. pp. 434-443
http://www.pitt.edu/~vanlehn/distrib/Papers/Bayesianapproach-Abs.html